REMARKS

In response to the Final Office Action dated December 7, 2006, claims 1, 16, and 18-21 are amended. Claims 1-21 are now active in this application. No new matter has been added.

Applicant appreciates that the Examiner, at page 2 of the Office Action, has withdrawn the objection to the specification, the rejection under 35 U.S.C. § 101, and the rejection under 35 U.S.C. § 112.

Independent claims 1, 2, 7, 11, 14-16, and 18-21 were rejected under 35 U.S.C. § 102(e) as allegedly anticipated by Kahle et al. (U.S. Patent Publication 2005/0044434). Claims 3, 4, and 7 were rejected under 35 U.S.C. § 103(a) as being allegedly unpatentable over Kahle in view of "Unifying FPGAs and SIMD Arrarys" by Bolotski et al. Claims 5 and 6 were rejected under 35 U.S.C. § 103(a) as being allegedly unpatentable over Kahle in view of Fuller (U.S. Patent 5,423,051). Claims 8-10, 12, and 13 were rejected under 35 U.S.C. § 103(a) as being allegedly unpatentable over Kahle in view of *In re Rose*, 220 F.2d 449, 105 USPG 237 (CCPA 19955).

Independent claim 1 recites, in pertinent part, "wherein the decode unit is operable to detect for each instruction packet whether the instruction packet defines (i) a plurality of only control instructions to be executed sequentially on the first processing channel or (ii) a plurality of instructions comprising at least one data processing instruction to be executed simultaneously on the second execution channel, and to control the first and second channels in dependence on said detection." Independent claims 18-21 recite similar limitations.

Anticipation under 35 U.S.C. § 102(e) requires that each element of the claim in issue be found, either expressly described or under principles of inherency, in a single prior art reference, *Kalman v. Kimberly-Clark Corp.*, 713 F. 2d 760, 218 USPQ 781 (Fed. Cir. 1983). At a

minimum, the cited prior art does not disclose (expressly or inherently) the above recited limitation.

The Office Action, at page 2, asserts that "it is inherent that instructions are issued to the proper functional unit based on opcode." However, in relying upon the theory of inherency, "the examiner must provide a basis in fact and/or technical reasoning to reasonably support the determination that the allegedly inherent characteristic necessarily flows from the teachings of the applied prior art." *Ex parte Levy*, 17 USPQ2d 1461, 1464 (Bd. Pat. App. & Inter. 1990) (emphasis in original).

The Office Action, at page 13, asserts that the last sentence of paragraph 54 of Sharangpani et al. (USPGPub 2004/0215593) comments that instructions are routed to a particular execution unit based on opcode. The Office Action further asserts, "that this is such a common aspect of computer architecture that it is seldom mentioned. However, logic will dictate, for example, that an integer unit is not capable of executing a floating-point instruction. A decoder is able to tell if an instruction is a floating-point instruction based on the opcode, and therefore routes the instruction to a particular execution unit based on the opcode. Further, the two instances of the contents of the packets mentioned in the claim may be two of many possible combinations that Kahle may handle. This may or not be the same intension as the applicant. If the claim is limited to ***only*** these two situations, Kahle may not be able to teach the limitation." Emphasis in original.

Specifically, Sharangpani, at paragraph 54, states, "a floating point unit and an integer unit are execution units in a general purpose processor. Based on the type of instruction (e.g., opcode), a decoder in the general purpose processor portion is able to direct instructions to either of these execution blocks." According to Microsoft Computer Dictionary, Fifth Edition, an

operation code (opcode) is the "portion of a machine language or assembly language instruction that specifies the type of instruction and the structure of the data on which it operates." A complete machine instruction contains an opcode, and, optionally, the specification of one or more operands.

Sharangpani merely uses the opcode to **route a single instruction** to either a floating point unit or an integer unit inside of a general purpose processor. Sharangpani does not address instruction packets containing multiple instructions.

Additionally, Kahle merely discloses instruction packets having control **and** data instructions. These packets are decoded in the decoder block 14 of FIG. 1 of Kahle, and the data instructions sent to block 20, 22, or 24, and the control instructions sent to block 26. It is not possible for Kahle to have an instruction packet comprising only control instructions.

In contrast to Kahle, and in contrast to Sharngpani, claim 1 detects an instruction packet containing multiple instructions, and determines whether the instruction packet contains "(i) a plurality of only control instructions to be executed sequentially on the first processing channel" or "(ii) a plurality of instructions comprising at least one data processing instruction to be executed simultaneously on the second execution channel." This determination requires **evaluating a group (packet) of instructions**, and not merely evaluating a single instruction as disclosed by Sharngpani.

Additionally in contrast to Kahle and Sharngpani, claim 1 also controls "the first and second channels in dependence on said detection." As an example of claim 1, see Applicant's specification at Figure 2 and paragraph [0027], wherein a value of "0" in the first bit of a packet of instructions indicates an opportunity for simultaneous or parallel execution.

The Office Action also asserts, at pages 13 and 14, that "[t]he mere presence of a vector unit shows that Kahle is capable of executing multiple instructions simultaneously." However, it is well known that **a vector unit is only capable of executing one instruction simultaneously** on a plurality of data, and is not capable of simultaneously executing a plurality of different instructions. Therefore, it is clear the Kahle is not capable of executing multiple instructions simultaneously. Consequently, even if Kahle were amended to process packets comprising only control instructions, which is not trivial, the resulting processor would not be able to perform simultaneous execution of instructions.

As recited by claim 1, the use of instruction packets comprising only control instructions which are dealt with by a first processing channel, and instructions packets comprising at least one data instructions which is dealt with by a second processing channel, enables claim 1 to process instructions more quickly than the prior art such as Kahle.

An example of claim 1 is discussed at pages 7 and 8 of the present application, and the example is illustrated by FIGs. 1-3. An instruction packet 100 may be one of three types: 1) instruction packet 211 which contains only control instructions; 2) instruction packet 212 which contains a data instruction and a memory instruction; and 3) instruction packet 213, which contains a data instruction and a control instruction. When a packet of type 211 is decoded, all of the control instructions contained in the packet are sent to the control execution unit 102 for processing; when a packet of type 212 is decoded, the data instruction and the memory instruction contained in the packet are sent to the data execution unit 103 for processing; and when a packet of type 213 is decoded, the data instruction contained in the packet is sent to the data execution unit 103 for processing and the control instruction contained in the packet is sent to the control execution unit 102 for processing. An advantage of this arrangement is that when

a first packet decoded is a control packet of type 211, then the instructions are all sent to the control execution unit. Then when the next packet decoded is a data packet of type 212 the instructions are all sent to the data execution unit for processing. **These two packets can then be processed simultaneously,** the control packet 211 in the control execution unit 102 and the data packet 212 in the data execution unit 103. This arrangement provides great advantage over the computer processors of the prior art in that the processing speed is greatly increased.

Thus, Applicant respectfully submits that independent claims 1, and 18-21 are patentable over Kahle, and are also patentable over the combination of Kahle and Sharngpani. The other cited publications (Bolotski, and Fuller) and cases (*In re Rose*) do not remedy the deficiencies of Kahle and Sharngpani.

Under Federal Circuit guidelines, a dependent claim is nonobvious if the independent claim upon which it depends is allowable because all the limitations of the independent claim are contained in the dependent claims, *Hartness International Inc. v. Simplimatic Engineering Co.*, 819 F.2d at 1100, 1108 (Fed. Cir. 1987). Accordingly, as independent claim 1 is patentable for the reasons set forth above, it is respectfully submitted that all claims dependent thereon (claims 2-17) are also patentable. In addition, it is respectfully submitted that the dependent claims are patentable based on their own merits by adding novel and non-obvious features to the combination.

For example, none of the cited prior art discloses or suggests the limitation of claim 16 which states, "wherein the decode unit is operable to read the values of a set of designated bits at predetermined bit locations in each instruction packet of the sequence, to determine: a) whether the instruction packet defines a plurality of only control instructions . . ." As an example of claim 16, see FIG. 2, instruction packet 211, wherein an initial bit of "1" indicates an instruction

packet with three control instructions. Thus, the instruction decoder 101 of FIG. 1 can identify an instruction packet with three control instructions merely by examining a single bit.

Thus, it is respectfully submitted that dependent claims 2-17 are allowable for, at a minimum, the same reasons as independent claim 1.

Accordingly, it is urged that the application, as now amended, is in condition for allowance, an indication of which is respectfully solicited. If there are any outstanding issues that might be resolved by an interview or an Examiner's amendment, Examiner is requested to call Applicants' attorney at the telephone number shown below.

To the extent necessary, a petition for an extension of time under 37 C.F.R. 1.136 is hereby made. Please charge any shortage in fees due in connection with the filing of this paper, including extension of time fees, to Deposit Account 500417 and please credit any excess fees to such deposit account.

Respectfully submitted,

McDERMOTT WILL & EMERY LLP

Ed Garcia-Otero

Eduardo Garcia-Otero
Registration No. 56,609

600 13th Street, N.W.
Washington, DC 20005-3096
Phone: 202.756.8000 SAB/EG:cac
Facsimile: 202.756.8087
Date: June 4, 2007

Please recognize our Customer No. 20277 as our correspondence address.

WDC99 1402928-2.066365.0020